arXiv:q-bio/0606023v2 [q-bio.QM] 20 Jun 2006

Subgraph Ensembles and Motif Discovery Using a New Heuristic for Graph Isomorphism

Kim Baskerville[1, 2] and Maya Paczuski[2]

[1]*Perimeter Institute for Theoretical Physics, Waterloo, Canada, N2L 2Y5*
[2]*Complexity Science Group, Department of Physics and Astronomy,*
University of Calgary, Calgary, Alberta, Canada T2N 1N4

(Dated: February 5, 2008)

A new heuristic based on vertex invariants is developed to rapidly distinguish non-isomorphic graphs to a desired level of accuracy. The method is applied to sample subgraphs from an E.coli protein interaction network, and as a probe for discovery of extended motifs. The network's structure is described using statistical properties of its N-node subgraphs for $N \leq 14$. The Zipf plots for subgraph occurrences are robust power laws that do not change when rewiring the network while fixing the degree sequence — although the specific subgraphs may exchange ranks. However the exponent depends on N. The study of larger subgraphs highlights some striking patterns for various N. Motifs, or connected pieces that are over-abundant in the ensemble of subgraphs, have more edges, for a given number of nodes, than antimotifs and generally display a bipartite structure or tend towards a complete graph. In contrast, antimotifs, which are under-abundant connected pieces, are mostly trees or contain at most a single, small loop. The extension to directed graphs is straightforward.

PACS numbers: PACS Numbers: 02.10.0x, 87.10.+e, 89.75.Fb

I. INTRODUCTION

The recent surge of interest in complex networks has often targeted general features of organisation [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. A number of common properties have been observed, including the so-called small world effect, fat tails in the distribution of the node degree (the "scale-free" network), as well as clustering. Although the last two attributes are statistical properties of the local network structure, networks that share these features may nonetheless exhibit totally different specific local structures. Certain connected subgraphs with three or four nodes, termed "motifs" [11, 12, 13], turn out to be significantly over-abundant in real networks when compared to null models. These null models are typically randomised networks where the smaller scale structure (e.g. node degree) [14] is determined by the original network. It is believed that networks with similar functions – for example, forward logic chips and neural networks – display the same motifs [11]. A growing body of evidence indicates that particular motifs perform specific functions in gene transcription networks [12, 15, 16, 17, 18, 19, 20, 21, 22]. In addition, proteins within motifs are more conserved across species than proteins that do not form part of such units [23, 24].

Motifs and antimotifs, which are significantly under-abundant connected subgraphs, may also be useful in classifying networks and comparing real-life situations to theoretical models. Milo *et al.* [25] explored significance profiles: normalised Z-scores for particular connected subgraphs. They claim to find "superfamilies" of networks displaying similar profiles. In a similar vein, Middendorf *et al.* [26] used exhaustive subgraph enumeration of networks generated by different theoretical models as training data for a machine learning algorithm, and developed a discriminative classifier subsequently able to identify new networks with success.

However, all of these approaches have been handicapped by the small size of connected subgraphs. This limits the scale where features of organisation in networks can be discovered. In most cases, connected subgraphs with at most four nodes are considered. Middendorf *et al.* [26] searched for two different categories of subgraphs: graphs which could be generated by a random walk of length less than or equal to eight, and graphs with up to seven links - to achieve slightly larger subgraphs. Ziv *et al.* [27] analysed statistically significant measures that can be calculated directly from the adjacency matrix. These measures are related to subgraphs but lack a one-to-one correspondence. Hence the possibility of insight into the function of organised structure at different scales or the systematic discovery of larger scale structures is - from our point of view - lost.

The existing size limitation for motif discovery leaves some interesting questions unanswered. Do motifs appear independently, or do they combine to form larger organised structures [27, 28, 29] that are overwhelmingly represented in the real network compared to an appropriate null model? If so, what do these extended structures look like? What properties of the network's ensemble of N-node subgraphs distinguish it from null models or from other networks? Are collections of nodes that participate in motifs of larger sizes also more likely to be related to function and/or conserved through evolutionary history? Kashtan *et al* made some progress in this direction by considering specific generalisations of three and four node motifs [30]. They found that networks sharing a particular three node motif favoured different generalisations of that motif, suggesting that larger structures need to be considered to fully understand how the network is organised. However, this work relied on *a priori* assumptions about possible generalisations to larger motifs. Searches were tailored to particular subgraphs. A more general analysis is known to be computationally difficult [31, 32, 33, 34].

A. Problems in Finding Extended Structures

There are at least three main problems. The first is that the time required for exhaustive enumeration of subgraphs increases rapidly with subgraph size, particularly for large networks. This can be solved by sampling: Kashtan et al [33] showed that quite small samples could be sufficient to identify motifs with up to seven nodes. However, their method requires the calculation of weights in order to achieve uniform sampling. Their calculation of these weights increases in difficulty, with combinatorial factors, as the the connected subgraph size increases. We achieve uniform sampling automatically by picking nodes at random from the network – at the expense of sampling both connected and disconnected subgraphs.

The second problem is to determine appropriate null model(s) and significance. The standard null model (see for instance Ref. [14]) is where the degree of every node is not allowed to change – so the single node properties are fixed. Such an ensemble can be obtained using a Sequential Monte Carlo method called "rewiring". Statistically significant deviations from that background are by definition coming from node-node correlations. Extending this argument, when Milo *et al.* [11] search for 4-node motifs they also fix the actual number of each kind of 3-node subgraph in their null model. However, as in Ref. [30], here we use only the ensemble of fixed degree sequence as a null model to test for significance. Explicitly fixing the occurrence of $(N-1)$-node subgraphs is computationally intractable for larger N. There are not only linear constraints between different subgraphs arising from conservation laws (see Ref. [25] and Section IV C) associated with rewiring but also non-linear correlations caused, in part, by the form of the null model.

The third difficulty lies in distinguishing non-isomorphic subgraphs. This is the well-known and notoriously difficult "graph isomorphism problem" [35, 36]. The number of possible graphs grows faster than exponentially with N [34]. Several algorithms [37, 38, 39, 40, 41] are available, but most of these are configured to make a comparison for isomorphism between two graphs. Comparing each new subgraph pairwise to all subgraphs already identified would be far too time-consuming in this context. Some existing programs can be altered to provide sets of labels to identify particular graphs. They tend to be optimised for large graphs (hundreds of nodes), and appear to us to be unsuitable for the type of search required for discovery of organisation at larger scales than three or four nodes.

At this point in time, discovery of larger scale organisation does not require particularly large subgraphs. Ten to fifteen nodes would already be a significant step forward, and entails a new set of problems and types of behaviours as discussed later. Subgraphs do, however, need to be classified quickly if a method is to be practical. We present a new heuristic that assigns a set of labels to each subgraph as it is sampled, so that isomorphic graphs are guaranteed to have the same label(s), but (most) non-isomorphic graphs have different labels. The accuracy of the method depends on the number of labels used – at the expense of increased computational effort. We test the heuristic by comparing with exact enumeration of all isomorphic graphs up to $N = 8$. Combined with a sampling technique, our heuristic is used to identify extended motifs of a protein interaction network. We sample both connected and disconnected subgraphs uniformly by picking N distinct nodes at random. Motifs are then discovered by looking at the significance – with some caveats – of individual subgraphs that contain these structures as distinct pieces.

B. Summary

The labelling algorithm is described in Section II. In Section III various stages of the algorithm are tested. The full algorithm successfully distinguishes all graphs with up to eight nodes [42]. Differences in the running times and accuracy of the stages are also discussed. In Section IV, the algorithm is used to identify extended motifs and antimotifs in the E. coli protein interaction network. The motifs all share a remarkably similar bipartite structure, which is completely different from the long chains and tails seen in antimotifs. For fixed N the distribution of all subgraph counts is found to obey a power law, where the exponent depends on N. However, the Zipf plots of the real and randomised networks are quite similar although the subgraphs exchange rank. In Section V we conclude with a summary.

II. THE LABELLING ALGORITHM

The algorithm developed here can be applied to both simple graphs and digraphs – graphs with directed edges. Here we will concentrate on the algorithm for simple graphs, leaving the straightforward generalisation to digraphs to a later publication.

Motif discovery requires a fast way to identify graphs that are isomorphic. One way to be certain that two graphs are isomorphic is to find the isomorphism that maps one to the other. This is a permutation of the vertex labels of one graph such that its list of links becomes identical to that of the other graph. To show that two graphs are not isomorphic therefore requires proving that no such isomorphism exists, which in theory requires checking every possible permutation of the vertices. Since there are $N!$ such permutations for a graph with N nodes, this is far too time-consuming to be practical. Many al-

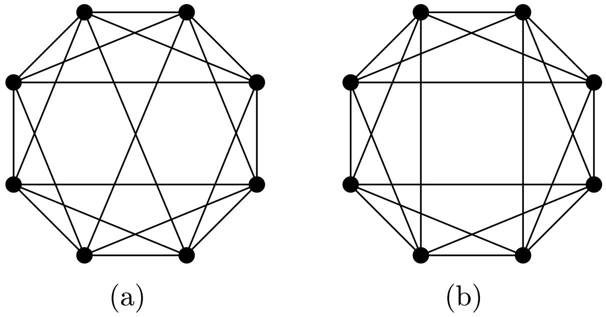
(a) (b)

FIG. 1: Two non-isomorphic graphs that cannot be distinguished by any of the invariants proposed by Remie.

gorithms therefore start by trying to reduce the number of permutations that need to be checked, usually by applying some kind of "canonical labelling" [2] or ordering to the vertices. For example, if a unique way of ordering the vertices in both graphs can be found, then vertices of the same rank must map to each other – in order for the graphs to be isomorphic.

An alternative approach is to try to find an invariant under permutation, or set of invariants, that uniquely labels any graph. The use of invariants ensures that isomorphic graphs always receive identical labels. However it is not certain that non-isomorphic graphs will receive at least one different label. Remie [43] defines four different invariants, but none of these can distinguish the eight node graphs in Fig. 1 as non-isomorphic.

A. Invariant Vertex Labels

Our approach defines vertex invariants through a generalisation of standard canonical labelling [2]. Usually, the canonical label depends only on the degrees of the vertex being labelled together with its immediate neighbours. This means, for example, that all vertices in a long chain (except the two endpoints) receive the same label, whereas it is clear that nodes near the end of the chain should be distinguishable from nodes nearer the middle. Bearing this in mind, we have extended the usual canonical labelling to include all vertices in the graph. In the case of a graph made of disconnected pieces, we include all vertices in the connected piece containing the vertex being labelled.

As with usual canonical labelling, our label is a sum of powers of two, with the vertex degrees, k_j, determining the power. To include all vertices, but give a higher weight to those closest to the vertex V_i being labelled, we include an additional factor of $2^{x-x_{ij}}$, where x is the diameter. This diameter is the maximum shortest path between any two vertices on the connected piece of the subgraph containing V_i. The quantity x_{ij} is the distance between vertices V_i and V_j, where V_j is required to be connected to V_i by some path. The lowest possible weighting is $2^0 = 1$ (if $x_{ij} = x$), and the highest weighting (2^x) is given to V_i itself. Each vertex V_i is assigned a label X_i as follows:

$$X_i = \sum_{j}^{\text{connected}} 2^{x-x_{ij}+k_j} \quad , \tag{1}$$

where k_j is the degree of vertex V_j [44]. The sum is taken over all vertices in the graph, or if the graph contains several disjoint subgraphs, over all vertices in the connected subgraph containing V_i.

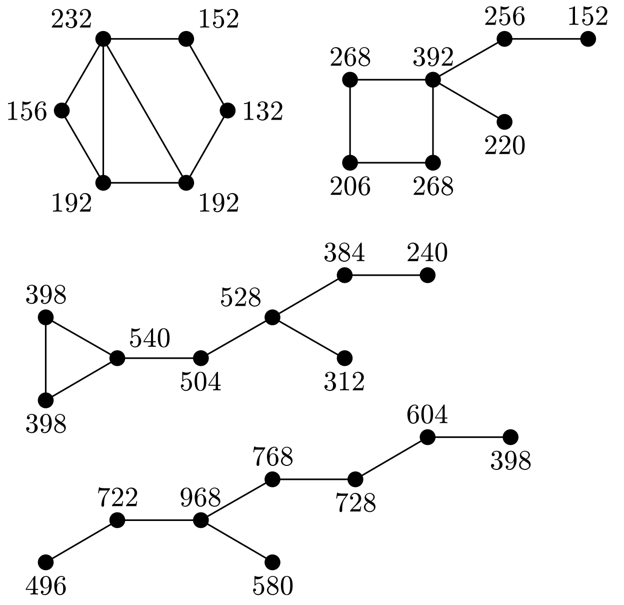

FIG. 2: Vertex labels calculated using Eq. (1). Higher values are assigned to more central vertices, or those with higher degrees.

The labels defined by Eq. 1 have an intuitive meaning. More connected or central vertices have higher values. Fig. 2 gives some examples of the labelling scheme for different subgraphs. The labels X_i are clearly higher for more central vertices than those closer to the edge.

B. Invariant Graph Labels

The set of vertex labels could be used directly to distinguish graphs, but they would need to be ordered, for instance in descending order, before comparisons between graphs could be made. Another approach is to combine the vertex labels to obtain a small set of graph labels. One candidate graph label is the sum $l_1' = \left(\sum_i X_i\right)$. Unfortunately it does not produce unique labels. Fig. 3 shows two graphs that have the same sum despite having different vertex labels (and hence being clearly non-isomorphic). However the product does not suffer from this defect. In theory it could, but in practice we have not found it to be the case for the graphs studied. Our

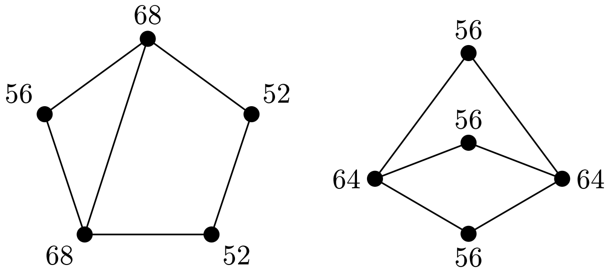

FIG. 3: These two graphs have different vertex labels X_i, which nonetheless combine to give the same sum: $l_1' = 68 + 68 + 56 + 52 + 52 = 64 + 64 + 56 + 56 + 56 = 296$. Their graph labels l_1, however, are not equal.

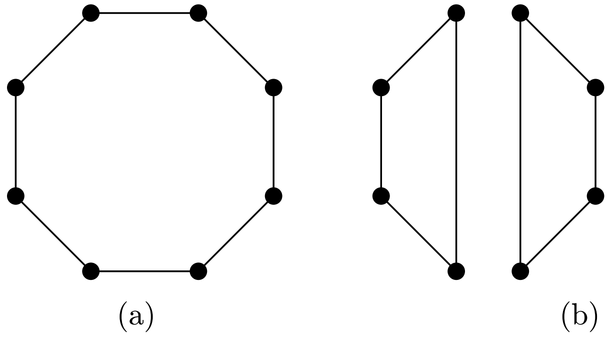

FIG. 4: These two graphs are the complements of the graphs shown in Fig. 1.

first graph label is therefore defined to be

$$l_1 = \prod_i X_i \quad . \tag{2}$$

Note that this product is over all the vertices in the graph, whether it is connected or made of disjoint pieces. Should this product become too large to be conveniently stored as an integer, the first several (eg. 9) digits can be used instead, without causing any degeneracy in labels. Again, this is an empirical observation rather than a mathematical certainty. However, this is not the end of the story.

We found that l_1 successfully distinguishes all graphs with up to five nodes, but there are two pairs of non-isomorphic graphs with six nodes that are assigned identical values. The graphs in Fig. 1 provide another problematic example. These graphs are highly symmetric. In both graphs, every vertex has degree five – with the remaining nodes at distance $x_j = 2$. Hence all the labels, $X_i = 2^2 * 2^5 + 5 * 2^1 * 2^5 + 2 * 2^0 * 2^5 = 576$, are identical.

If all vertices are equally "connected", but the two graphs are not isomorphic, what is the difference between them? Taking their complements (exchanging links and non-links for every vertex pair) as shown in Fig. 4 reveals the source. While the complement of graph (a) is a single loop with 8 links, which we shall now refer to as an 8-loop, that of graph (b) consists instead of two 4-loops. Applying our labelling method to these complements produces unique labels, which suggests a possible solution to the problem. For all graphs, first calculate l_1 as described above. Then take the complement of each disconnected subgraph of the graph. Recalculate labels, Y_i, for this new graph, and combine these labels into the product

$$l_2 = \prod_i Y_i, \tag{3}$$

where the product is again taken over all vertices in the graph. Each graph is then labelled by the vector (l_1, l_2, L, N), where L is the total number of links in the graph. Note that for disjoint graphs, it is extremely important to *take the complement of each connected subgraph individually*; if the complement of the whole graph is taken instead, small disconnected pieces can cause problems, so that degeneracy in labelling appears for quite small graphs. An algorithm with these graph labels was tested by applying it to every possible labelled graph for $N \leq 8$ and measuring the number of distinct sets of labels. This number was compared to the true number of non-isomorphic graphs. Those were determined using Polya's enumeration theorem. The algorithm uniquely labelled every graph with up to six nodes ($N = 6$), distinguished 1038 out of 1044 for $N = 7$ and 12078 out of 12346 for $N = 8$. Even for $N = 8$ almost 98% of distinct graphs were uniquely labelled.

What further invariant properties can be used as labels? Again, considering the complements in Fig. 4 provides a clue – their different loop structures. In fact the numbers of all loops except 3-loops are different for the two graphs in Fig. 1. We counted all the loops in a graph by searching through its adjacency matrix. The number of 3-loops (n_3), 4-loops (n_4) etc. can then be incorporated as extra labels, so that each graph is labelled by the vector $(l_1, l_2, L, n_3, n_4, ..., n_N)$. This adapted algorithm, when tested, correctly distinguished *all* graphs with up to $N = 8$ nodes. Exhaustive testing of graphs with more nodes is not worthwhile at present, as the program for $N = 9$ would run for more than a year on a present day standard laptop.

III. TESTING THE ALGORITHM

This section may be skipped by those primarily interested in motif discovery. As stated in Section II, all stages of the algorithm have been tested exhaustively for graphs with up to eight nodes. A simple graph with N nodes contains $L_{MAX} = \binom{N}{2} = N(N-1)/2$ vertex pairs. Thus L_{MAX} is the maximum possible number of links, and $2^{L_{MAX}}$ is the number of labelled graphs. An easy way to generate all labelled graphs is to cycle through the binary numbers between 0 and $2^{L_{MAX}} - 1$, loading their digits in order into the off-diagonal elements of an adjacency matrix. The labelling algorithm can then be successively applied to each matrix or graph. The accuracy of the algorithm can be evaluated by comparing the number of graphs correctly distinguished to the true number of non-isomorphic graphs, as determined by Polya's enumeration theorem. The results for different stages of the algorithm are shown in Table I. Note that since the labels are invariants, isomorphic graphs must be assigned the same set of labels. Thus it is not possible to overcount the number of distinct graphs. Undercounting is possible, however, since non-isomorphic graphs may nonetheless have similar enough structures to produce degenerate labels.

Table I shows that incorporating loop counting together with l_1 and l_2 is the most accurate method. However the cost in computing time is significant. On a standard laptop, for $N = 8$ it took four and a half hours to compute l_1 alone, six hours to compute l_1 and l_2, and twenty six hours for the full algorithm including loop counting. Using l_1 and l_2 without loop counting works perfectly up to $N = 6$, but then misses 6 graphs (0.6%) at $N = 7$ and 268 graphs (2.2%) at $N = 8$. The graphs shown in Fig. 5 are typical examples of pairs not distinguished by either l_1 or l_2. One graph can be mapped to the other by switching the endpoints or "rewiring" two links. The complements of the graphs share the same property; hence the degeneracy in l_2 as well as l_1.

Another possible route might be to omit l_2 when loop counting is included. Using l_1 plus loop counting works perfectly up to $N = 7$, but fails to distinguish two pairs of graphs at $N = 8$ (see Fig. 6). The danger, as with omitting loop counting, is that once an algorithm misses even a small percentage of graphs for some N, it misses more and more as N increases.

To summarise: The combination of l_1, l_2 and loop enumeration differentiates all non-isomorphic graphs with up to eight nodes. However, loop counting is very time consuming, and omitting it only causes around 2% of the $N = 8$ graphs to be degenerately labelled. With the above mentioned caveats we proceed with a subgraph census obtained by sampling a protein interaction network using the algorithm with l_1 and l_2, but without loops.

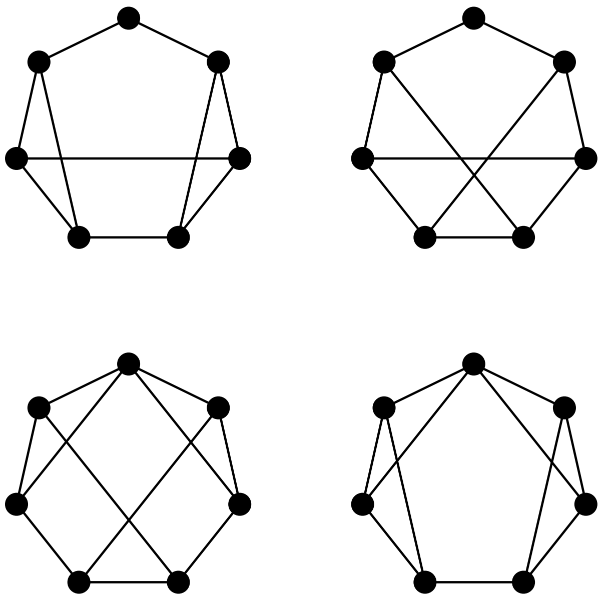

FIG. 5: The bottom pair of graphs are the complements of the top pair. Neither pair can be distinguished by l_1 or l_2. The pairs exhibit different loop structures and can therefore be differentiated by loop enumeration.

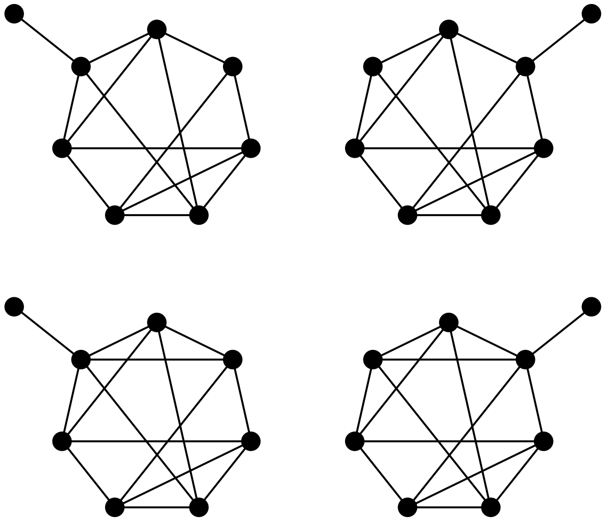

FIG. 6: The top and bottom pairs have the same vertex labels and the same loop structure. Both pairs are distinguished by the labels of their complements. Note that the bottom pair are identical to the top – save for the addition of one extra link.

TABLE I: Number of graphs distinguished by different graph labels compared to the exact number of graphs calculated using Polya's enumeration theorem, shown in the second column. The third column shows the result obtained by using the sum, l_1', rather than a product, l_1, of the vertex labels. In the remaining columns l_1 and l_2 are as defined in Equations (2) and (3). The last column includes the number of loops as graph labels.

	Number of Graphs				
N	**Exact**	$\mathbf{l_1'}$ **(sum)**	$\mathbf{l_1}$	$\mathbf{l_1, l_2}$	$\mathbf{l_1, l_2,}$ **loops**
2	2	2	2	2	2
3	4	4	4	4	4
4	11	11	11	11	11
5	34	33	34	34	34
6	156	136	154	156	156
7	1044	693	1004	1038	1044
8	12346	4381	11188	12078	12346

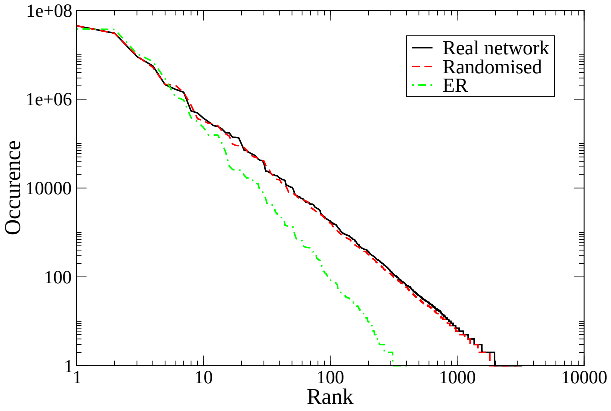

FIG. 7: Zipf plot for N=9 subgraphs of the giant component of the E. coli network, for sample size 10^8. Also shown are the Zipf plots for the rewired network and for a Bernoulli or Erdös-Renyi (ER) random graph with the same link probability and same sample size. Fixing the degree sequence almost exactly fixes the Zipf plot while the specific subgraphs exchange rank under rewiring.

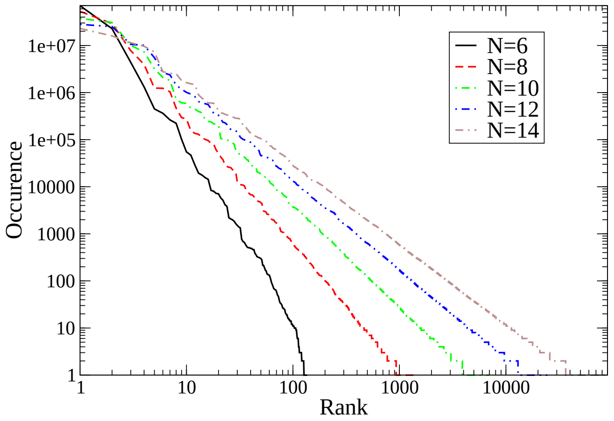

FIG. 8: Zipf plots obtained from the giant component of the E. coli network for subgraphs with varying numbers of nodes N.

IV. ENSEMBLES OF SUBGRAPHS AND MOTIF DETECTION IN A PROTEIN INTERACTION NETWORK

We now present results for the statistics of subgraphs in the protein interaction network of E. coli [45]. The histogram of all non-isomorphic subgraphs in the network is a characterisation of that network. This is termed a "subgraph census" [46]. The ensemble of subgraphs is obtained by uniform sampling rather than exact enumeration. This should give an accurate picture of the true census up to statistical fluctuations and an overall normalisation. Uniform sampling of connected and disconnected N-node subgraphs is achieved by picking N nodes at random. Results were compared with exact enumeration for small N. Since there is no inherent directionality in the interactions themselves, we have chosen to treat the network as undirected. The network has 270 nodes and 716 links; however it is not fully connected: seventeen pairs of nodes connect only to each other, and there are two isolated triplets. The largest connected component consists of 230 nodes and 695 links. Both this piece, termed the giant component (GC), and the entire network are studied.

A. Zipf's Law for Subgraph Census

We first consider subgraphs with a fixed number of nodes and ask what is the frequency of occurrence of different subgraphs. For each $N > 5$ a sample of 10^8 subgraphs were obtained. The ensembles for $N = 3$ and $N = 4$ do not have enough subgraphs to obtain a smooth distribution. The labels L, l_1 and l_2 were used to identify graphs, but loop counting was not included. The subgraphs were then ranked in descending order of occurrence, and Zipf plots were made [47].

The Zipf plots all indicate power law behaviour. Figure 7 shows a typical example. The distribution obtained from the GC was compared to two different null cases. The first, denoted "randomised" in Fig. 7, is a rewired version of the GC with the degree of each node fixed. This was generated by repeatedly choosing two links in the network at random and swapping their endpoints, until mixing was achieved. As usual, mixing was evaluated *a postiori*. Swaps are disallowed if they create self-loops or produce a pre-existing link. The second null model is a random Erdös-Renyi (ER) network with the same link probability as the real network. For the GC of the E. coli network, this link probability is $p = 695/\binom{230}{2} \approx 0.0264$, for the original network $p = 716/\binom{270}{2} \approx 0.0197$. An ensemble of 10^8 graphs with the desired number of nodes was generated using a Bernoulli process. In particular, a random number was placed on each pair of distinct nodes to determine whether or not a link would be made. This ensemble is denoted "ER" in the Zipf plots. As demonstrated in Fig. 7, the Zipf plots of the real and randomised networks are almost identical, but differ noticeably from the ER network. This is true for all N, and for both the GC the entire E. coli network.

Figure 8 shows Zipf plots for the GC with varying subgraph sizes. It can be seen that all five sizes are consistent with power law behaviour, although $N = 6$ is less smooth than the others because there are fewer subgraphs. The main difference between the Zipf plots is that as N increases the gradient becomes shallower. Hence, it appears that the exponent is not universal with respect to N.

Zipf plots for the original network and its GC are also similar. As Fig. 9 shows, the plots for the real network and the randomised network with identical degree sequence are close in both cases. The main difference between the GC and the entire network is that in the latter case the distribution is somewhat broader. However, the

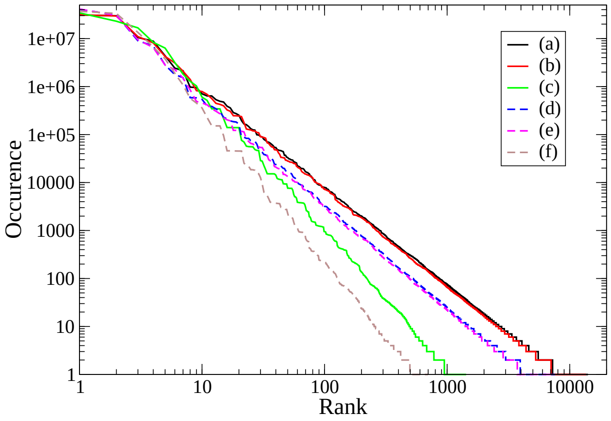

FIG. 9: Zipf plots for $N = 11$ subgraphs in different networks: (a) real E. coli network, (b) rewired E. coli network, (c) ER network with same number of nodes and link probability as (a), (d) giant component of E. coli network, (e) rewired giant component, and (f) ER network with same number of nodes and link probability as (d).

curves for the ER networks with corresponding link probabilities show the same tendency, which suggests that the difference in link probability may be the main factor for this trend.

B. Evidence for Motifs

Although the collection of subgraph counts are almost identical for the real and randomised networks, the rank of individual subgraphs within each census differs markedly. The subgraphs of the randomised network were arranged in the same order as those in the real network to get the scatter plot shown in Fig. 10. For comparison, the Zipf plot for the real network is also shown as a connected line. The vertical difference between each point and the line indicates the difference in the number of occurrences of a particular subgraph in the randomised network as compared to the original one. Note that the rank of the subgraph in the original network gives a unique tag to that subgraph. It can clearly be seen that the counts of certain individual subgraphs vary by orders of magnitude between the two networks.

These large differences are not just a statistical artifact of the rewiring process, as can be seen by re-doing the Fig. 10 for two randomised networks with the same degree sequence as the E. coli network. Now the subgraphs are ordered according to their occurrence in the first randomised network. Comparing Figs. 10 and 11, note that the scatter of points around the line (particularly below the line) in the latter case is significantly less than the former. This suggests the existence of “motifs” [11, 12, 13]: particular subgraphs that are significantly over-abundant in the real network compared to its ensemble of randomised networks.

To explore the issue of motifs further, subgraph counts

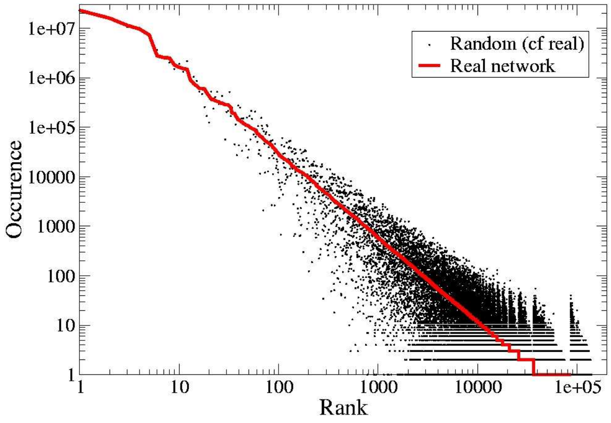

FIG. 10: Occurrences of $N = 14$ subgraphs for the real (red line) and randomised (black points) networks. The subgraphs in the randomised network are placed along the x-axis in the same order as those in the real network to allow direct comparison between counts for each subgraph. Points significantly below the line represent motifs, while those significantly above represent anti-motifs.

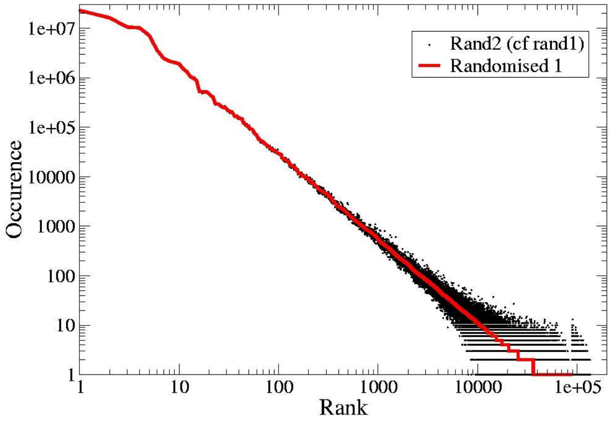

FIG. 11: This graph compares two randomised versions of the E. coli network in exactly the same way that the network and a randomised version of it were compared in Fig 10. The fluctuations, or scatter below and above the line in Fig. 10 are much larger, indicating a pattern of statistically significant deviations of subgraph occurrences in the original network.

from the real network were compared to counts from several randomised networks. For $N = 3$ and $N = 4$, we made an exhaustive enumeration of every subgraph. This was done for the real network and one hundred different randomised networks. The mean and standard deviation of the randomised counts were then computed, allowing a Z-score to be calculated. Fig. 12 shows the results for the original network (with Z-scores for the GC in brackets). The counts in the ER column are theoretical expectation values for an ER network of the appropriate size and link probability.

C. Linear Constraints Between Subgraphs

For $N = 3$ all the Z-scores all have the same magnitude. This is a direct consequence of the strict conservation of the degree sequence in the rewiring procedure [46]. Consider a particular swap between two links. The only 3-node graphs that can possibly be affected are those that contain at least one of the newly created or newly deleted links. At least two of the three nodes must therefore be chosen from among the four at the ends of the swapped links. The 4-node graph formed by the swapping nodes themselves is always unchanged by all allowed swaps (recall that it is not permitted to duplicate a pre-existing link). Its 3-node subgraphs are therefore also unaffected. The remaining possibility involves 5-node graphs containing one extra node in addition to the four swapping nodes. This extra node can have between zero and four links connecting it to the four swapping nodes. It turns out that there are only three pairs of 5-node graphs that can be interchanged by link swapping. In every case, the count of $N = 3$ graphs with no links decreases by one, that of one-link graphs increases by three, that of two-link graphs decreases by three and that of three-link graphs increases by one (up to an overall sign). This exact equality produces coincidence of the Z-scores.

The only remaining degree of freedom for the deviation of the actual network from its randomised ensemble is a single signed number. Its value indicates a significant difference between the real network and random networks with the same degree sequence, although it is impossible to ascribe this significance to any one subgraph in particular. Note that for the empty 3-node graph, its statistical under-abundance in the real network is due to the fact that the variance of this number in the ensemble is tiny, because those changes are slaved to a variable (the connected triangle) with small numbers. The actual under-abundance of empty 3-node subgraphs is an unimportant fraction of the overall number of those subgraphs. This illustrates the potential difficulties with assigning importance to individual subgraphs based on their individual Z-score – when the Z-scores must be correlated.

Conservation rules for subgraphs under rewiring was previously observed by [25] for 3-node subgraphs in directed networks, where although there are thirteen different connected motifs, only seven degrees of freedom are independent. For undirected N-node graphs, there are N conservation laws corresponding to moments of $\sum_i k_i^m$ with $m = 0, 1 \ldots N - 1$. Hence for $N = 3$ there is only one independent degree of freedom while for $N = 4$ there are seven.

D. Motif Selection

Ignoring the potential problems associated with attaching physical importance to specific subgraphs with high individual Z-scores, we find that for $N = 4$ two graphs stand out as being particularly over- or under-abundant. The square graph labelled $l_1 = 1679616$ is over-represented, while the same graph with one edge missing ($l_1 = 6350400$) is under-represented. It is also interesting to note that graphs with more (less) links tend towards over (under)-abundance. Overall the Z-scores are modestly lowered for the GC, but the same overall trends emerge in both cases. In particular, the same two subgraphs are readily identified as motif and anti-motif.

For $N \geq 5$, an exhaustive scan of all subgraphs is time-consuming, so uniform samples of 10^8 subgraphs were used instead. Subgraphs do not need to be fully connected in order to be useful for identifying motifs. As for $N = 3$ and $N = 4$, the real network was compared to an ensemble of randomised networks with the same degree sequence. Only 20 networks were included, though, rather than 100. Twenty was chosen as the smallest number for which standard deviations and Z-scores are reasonably stable. Checks show that when calculations are repeated, the Z-scores obtained vary slightly, but the same graphs always stand out as motifs.

The main difficulty is that too many subgraphs have high individual Z-scores. This may be related to the correlations discussed above. Ignoring previously mentioned caveats, we proceeded by using other selection criteria to choose the most important. After some experimentation the following *ad hoc* rules were used to identify motifs. Two different samples were taken from the real network, and Z-scores were computed comparing each of these to the same ensemble of 20 randomised networks. A subgraph was identified as either a motif (if it was connected) or containing a motif (if it was disconnect) if $Z > 10$ (or $Z < -10$ for anti-motifs) for both samples. Note that we only consider connected pieces to be motifs even though the subgraphs from which motifs are identified may be disconnected. Requiring $|Z| > 10$ for two different samples largely eliminates statistical oddities, which can otherwise occur for subgraphs with low counts. The relatively high cut-off in Z also helps ensure statistical stability, as was also noted in [33]. Even then, the number of new motifs identified increases dramatically with N. To overcome this problem, only subgraphs whose Z-scores were in the top fifty for that value of N were considered. Again, this had to be true for both samples. Motifs identified at a given N tend to reappear as connected components in disconnected graphs at higher N – see for example the graph labelled 4096(1) for $N = 3$ and $N = 4$ in Fig. 12. The last condition was therefore that a new motif has to replace an old one in the top fifty to make the grade.

Since including extra, unconnected nodes does not change the label of a graph it easy to identify and eliminate previous motifs at each new value of N. Motifs with a given number of nodes are not always discovered straight away; for example an $N = 6$ motif may not meet the condition $Z > 10$ in the sample of $N = 6$ subgraphs but show up much more strongly (with one or two disconnected nodes) at $N = 7$ or $N = 8$. This often means that subgraphs which only just fail the criteria at one N are positively identified at the next. This trend makes the selection of motifs more robust against small changes in the rules used to identify motifs. At some point, however, the number of genuine new motifs found begins to account for a smaller and smaller proportion of newly identified subgraphs. We also found that for $N > 9$ a smaller proportion of sampled subgraphs had $|Z| > 10$. Because of these diminishing returns, the present search was stopped after $N = 10$.

There are several possible reasons for this loss of efficiency: one is the finite size of the E.coli network, or another property of the network. It is also possible that the heuristic may be starting to fail, recalling that the most accurate version was not employed because of time constraints. Wrongly classifying a small percentage of nonisomorphic graphs as isomorphic is unlikely to make much difference, but if the problem worsened, genuine motifs could be swamped by other subgraphs which are more common in the randomised networks. This potential difficulty does not cast doubt on the motifs or antimotifs presented here, as none of them fall into the categories of graphs that cause problems, which have been thoroughly investigated for $N \leq 8$. However, further investigation might be appropriate before attempting to use this method for much larger subgraphs.

The original network was considered first, then calculations were repeated on the GC of the network the first few N. The same motifs were identified for both networks, although the order in which they were found varied slightly. We therefore conclude that the technique is robust.

E. Patterns in Motifs

The motifs found are shown in Figs. 13 (over-abundant) and 14 (under-abundant). Some striking patterns appear. First, many of the motifs have a bipartite structure where the vertices can be divided into two sets such that no links exist within either set, but many links exist between members of opposite sets. Many graphs display a complete matching: each vertex is connected to every member of the other set. Many more graphs have almost complete matchings, missing just one or two links. Again, some graphs are almost bipartite, with complete or almost complete matchings between two sets of vertices, and just a few matchings within each set. Some of these latter graphs may be seen as interpolating between bipartite graphs and complete graphs, where every vertex connected to every other vertex. Complete graphs at $N = 4$ and $N = 5$ are observed as motifs. All motifs have a high link:node ratio. In fact, $L \geq N$ for all motifs. Finally, the remaining motifs fall into one of the categories described above, with the addition of one or two "hanging" links.

Antimotifs follow a completely different pattern. They occur mostly as trees or may contain at most a single loop (usually a triangle, but there are two pentagons and one square) with long tails. This is to be contrasted with the bipartite structures of the over-abundant subgraphs, which typically contain many loops. They also have fewer links than motifs: either $L = N-1$ for pure chains or $L = N$, if there is one loop). This difference in the link:node ratios is readily apparent in Table II. In fact, for a given N no overlap in L values for motifs and antimotifs exists.

V. SUMMARY

This paper addresses some of the problems associated with finding extended structures in complex networks. We propose a new heuristic for graph isomorphism and validate its accuracy for classifying all undirected subgraphs with N up to 8. A version of the algorithm is used, together with uniform sampling, to obtain statistical signatures of the ensemble of N-node subgraphs in an E. coli protein interaction network for subgraphs with N up to 14. The distribution of subgraph occurrences follows a power law and the Zipf plots do not change significantly under rewiring. Sampling all possible subgraphs for various N allows for the discovery of extended motifs. Motifs are considered to be individual, connected graphs that are vastly over-represented in the network compared to a null model. They have more edges, for a given number of nodes, than antimotifs and generally display a bipartite structure or tend towards a complete graph. In contrast, antimotifs, are mostly trees or contain at most a single, small loop. The heuristic for graph isomorphism developed here can be applied with minor changes to directed graphs.

Acknowledgments

We thank Peter Grassberger for enlightening discussions about problems in sampling and significance estimation. M. P. thanks Lee Smolin and the Perimeter Institute for their hospitality during the initiation of this research, and Stefan Boettcher for conversations about Zipf plots and motif discovery.

TABLE II: The number of motifs (bold) and antimotifs (italics) with a given number of nodes, N, and links, L. The two classes are separated in this space, and do not overlap.

	L										
N	3	4	5	6	7	8	9	10	11	12	13
4	*1*	**1**	**1**	**1**							
5		*2*	**1**	**1**	**3**	**2**	**1**	**1**			
6			*5*	*4*	**4**	**4**	**5**	**4**	**3**	**2**	**1**
7				*7*	*10*	**2**	**5**	**5**	**4**	**3**	
8					*4*	*4*			**1**	**1**	

Graph label	**Graph**	**Real**	**Randomised network**			**Bernoulli**
$\mathbf{l_1(l_2)}$		**network**	**Mean**	**Std. dev.**	**z**	
N=3:						
1 (1)		3061013	3061168.86	21.53	-7.24 (-5.18)	3056010
36 (1)		174844	174376.42	64.58	+7.24 (+5.18)	184396
7776 (36)		7805	8272.58	64.58	-7.24 (-5.18)	3709
4096 (1)		478	322.14	21.53	+7.24 (+5.18)	25
N=4:						
1 (1)		193364479	193404007.84	6196.33	-6.38 (-4.08)	192159278
36 (1)		20997994	20888180.84	17303.95	+6.35 (+3.99)	23189310
7776 (36)		1794103	1894318.20	14425.39	-6.95 (-4.60)	932810
1296 (1)		162396	156355.15	1008.98	+5.99 (+8.30)	233202
4096 (1)		109244	70539.76	4648.79	+8.33 (+6.29)	6254
6350400 (6350400)		54613	71653.91	953.80	-17.87 (-17.49)	18762
965888 (4096)		42464	45374.38	1135.46	-2.56 (-0.89)	6254
2935296 (7776)		12612	13158.36	788.40	-0.69 (-2.83)	377
1679616 (1296)		5785	1650.39	100.27	+41.23 (+35.08)	94
11075584 (36)		2425	1055.71	149.76	+9.14 (+7.16)	4
2560000 (1)		230	50.46	16.22	+11.07 (+7.86)	0

FIG. 12: Results for subgraphs with $N = 3$ and $N = 4$ nodes in the E. coli protein interaction network. The third column shows the counts obtained by exact enumeration for the real network, while columns 4-6 show results obtained from exact enumeration of subgraphs in an ensemble of 100 networks with the same degree sequence. Standard deviations for the giant component are shown in brackets in column 6. The last column shows theoretical expectation values for ER random graphs.

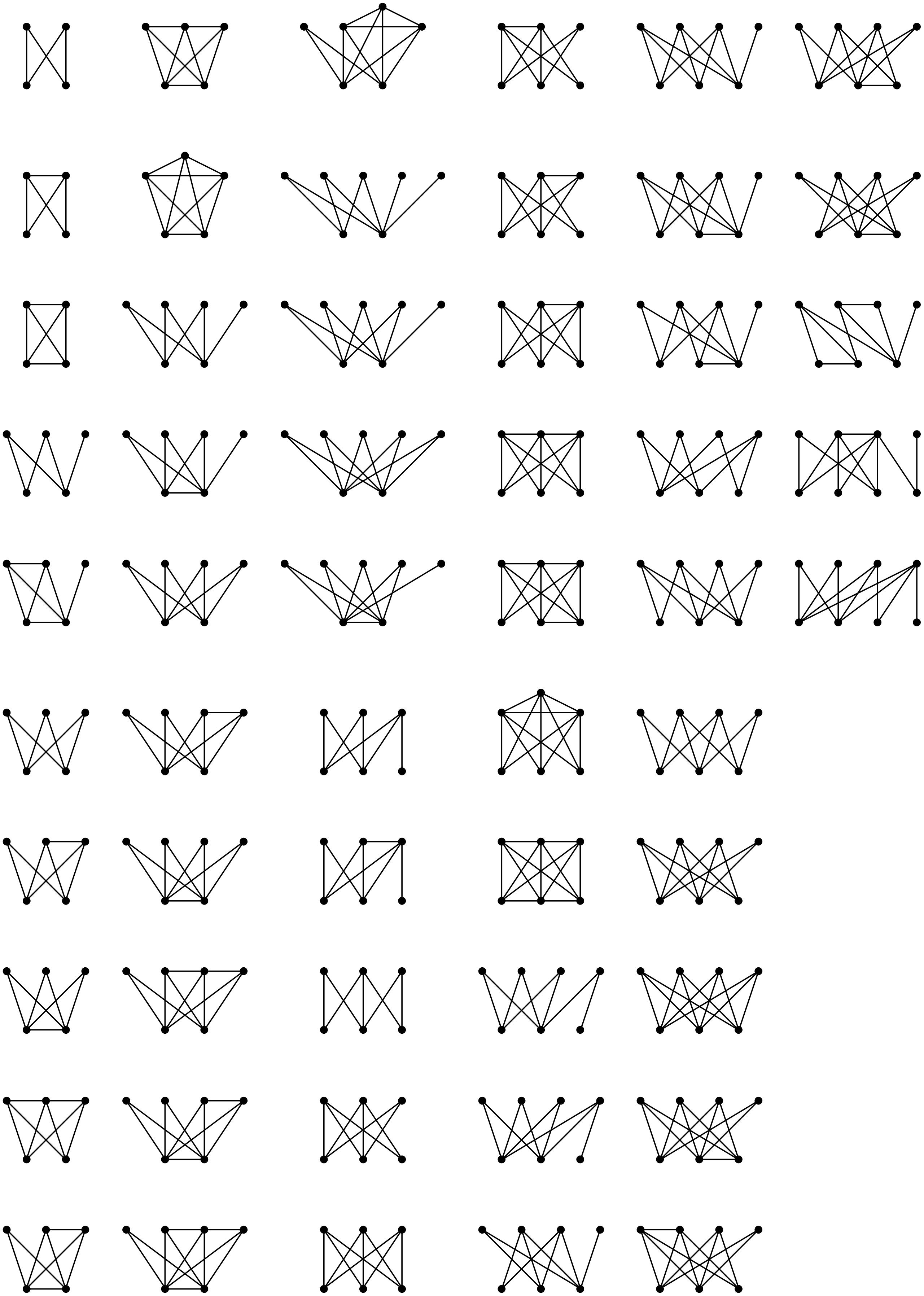

FIG. 13: Motifs (over-abundant subgraphs) of the E.coli protein interaction network.

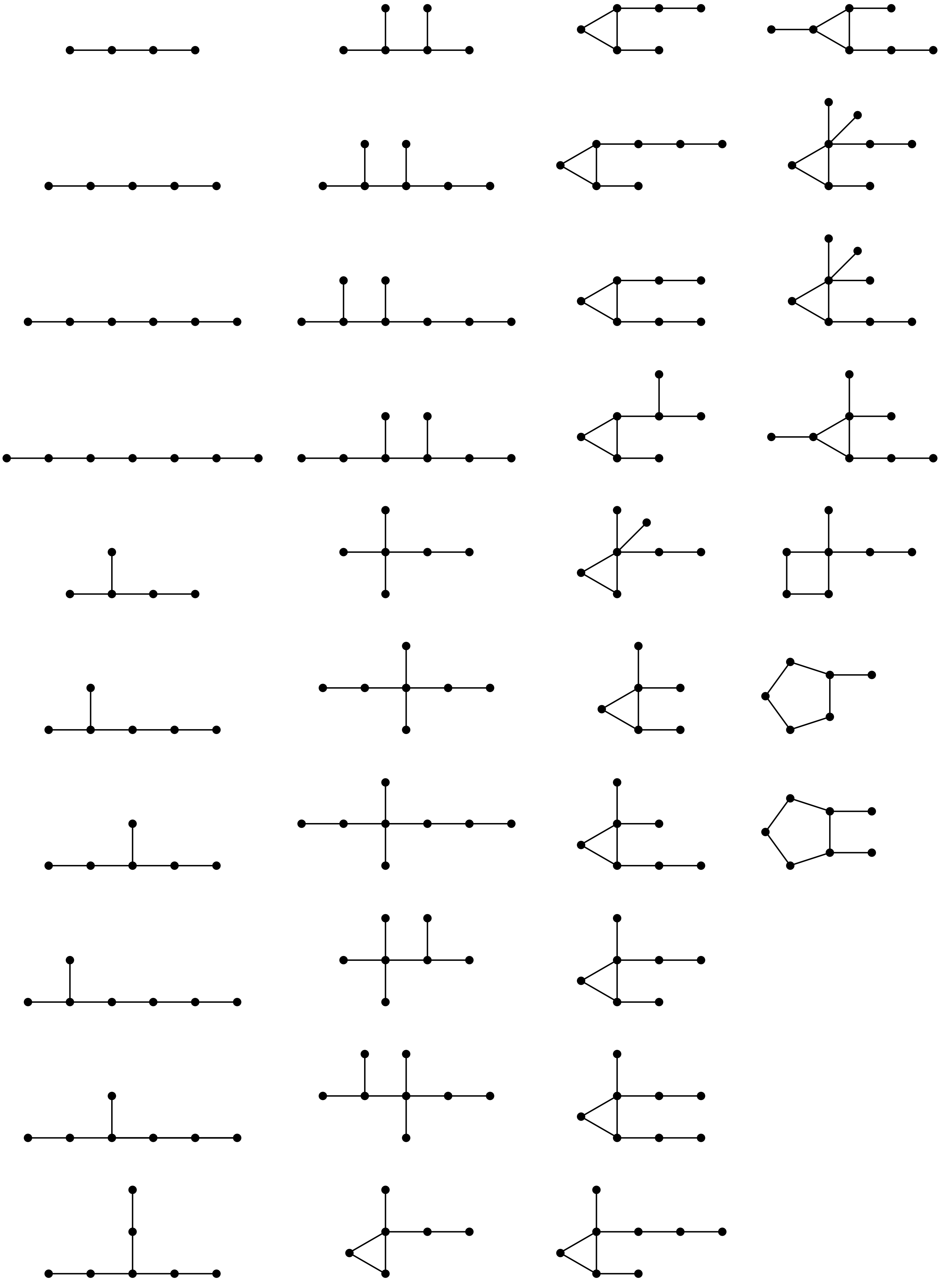

FIG. 14: Antimotifs (under-abundant subgraphs) of the E.coli protein interaction network.

---

[1] S. H. Strogatz, Nature **410**, 268 (2001).
[2] B. Bollobas, *Random Graphs* (Academic, London, 1985).
[3] D. Watts and S. Strogatz, Nature **393**, 440 (1998).
[4] A.-L. Barabási and R. Albert, Science **286**, 509 (1999).
[5] M. Newman, Proc. Natl. Acad. Sci. U.S.A. **98**, 404 (2001).
[6] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai and A.-L. Barabási, Nature **407**, 651 (2000).
[7] R. F. Cancho, C. Janssen and R. V. Sole, Phys. Rev. E **64**, 046119 (2001).
[8] R. F. Cancho and R. V. Sole, Proc. R. Soc. London Ser. B **268**, 2261 (2001).
[9] L. Amaral, A. Scala, M. Barthelemy and H. Stanley, Proc. Natl. Acad. Sci. U.S.A. **97**, 11149 (2000).
[10] B. Huberman and L. Adamic, Nature **401**, 131 (1999).
[11] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii and U. Alon, Science **298**, 824 (2002).
[12] S. Shen-Orr, R. Milo, S. Managan and U. Alon, Nat. Genet. **31**, 64 (2002).
[13] A. Vásquez, R. Dobrin, D. Sergi, J.-P. Eckmann, Z. N. Oltvai and A.-L. Barabási, Proc. Natl. Acad. Sci. U.S.A. **101**, 17940 (2004).
[14] S. Maslov and K. Sneppen, Science **296**, 910 (2002).
[15] N. Rosenfeld, M. B. Elowitz and U. Alon, J. Mol. Biol **323**, 785 (2002).
[16] S. Mangan and U. Alon, Proc. Natl. Acad. Sci. U.S.A. **100**, 11980 (2003).
[17] S. Mangan, A. Zaslaver and U. Alon, J. Mol. Biol. **334**, 197 (2003).
[18] M. Ronen, R. Rosenberg, B. I. Shraiman and U. Alon, Proc. Natl. Acad. Sci. U.S.A. **99**, 10555 (2002).
[19] A. Zaslaver, A. Mayer, M. Surette, R. Rosenberg, P. Bashkin, H. Sberro, M. Tsalyuk and U. Alon, Nat. Genet. **36**, 486 (2004).
[20] C. H. Yuh, H. Bolouri and E. H. Davidson, Science **279**, 1896 (1998).
[21] N. Buchler, U. Gerland and T. Hwa, Proc. Natl. Acad. Sci. U.S.A. **100**, 5136 (2003).
[22] Y. Setty, A. E. Mayo, M. G. Surette and U. Alon, Proc. Natl. Acad. Sci. U.S.A. **100**, 7702 (2003).
[23] S. Wuchty, Z. N. Oltvai and A.-L.Barabási, Nat. Genet. **35**, 176 (2003).
[24] A. Vespignani, Nat. Genet. **35**, 118 (2003).
[25] R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer and U. Alon, Science **303**, 1538 (2004).
[26] M. Middendorf, E. Ziv and C. H. Wiggins, Proc. Natl. Acad. Sci. U.S.A. **102**, 3192 (2005).
[27] E. Ziv, R. Koytcheff and C. H. Wiggins, e-print cond-mat/0306610.
[28] R. Dobrin, Q. K. Beg, A.-L.Barabási and Z. N. Oltvai, BMC Bioinformatics **5**, 10 (2004).
[29] J. Berg and M. Lassig, e-print cond-mat/0308251.
[30] N. Kashtan, S. Itzkovitz, R. Milo and U. Alon, Phys. Rev. E **70**, 031909 (2004).
[31] S. Itzkovitz, R. Milo, N. Kashtan, G. Ziv and U. Alon, Phys. Rev. E **68**, 026127 (2003).
[32] J. Nesetril and S. Poljak, Comments Math. Univ. Carol. **26**, 415 (1985).
[33] N. Kashtan, S. Itzkovitz, R. Milo and U. Alon, Bioinformatics **20**, 1476 (2004).
[34] F. Harary and E. M. Palmer, *Graphical Enumeration* (Academic Press, NY, 1973).
[35] J. U. Köbler and J. T. Schöning, *The Graph Isomorphism proble: Its Structural Complexity* (Birkhauser, Boston, 1993).
[36] J. Torán, FOCS 180 (2000).
[37] L. P. Cordella, P. Foggia, C. Sansome and M. Vento, *Proc. of the 10th International Conference on image Analysis and Processing* p.1172 (1999).
[38] P. Foggia, C. Sansome and M. Vento, *3rd IAPR-TC-15 International Workshop on Graph-based Representations* (2001).
[39] B. D. McKay, Congressus Numerantium **30**, 45 (1981).
[40] D. C. Schmidt and L. E. Druffel, J. Ass. Comp. Mach. **23**, 443 (1976).
[41] J. R. Ullman, J. Ass. Comp. Mach. **23**, 31 (1976).
[42] It is not known to fail for larger graphs, but exhaustive testing is not practical, nor does it appear to be necessary for our purposes.
[43] V. Remie, Bachelors Thesis, Eindhoven University of Technology (2003).
[44] Different bases could be used in the definition of X_i, for example $X_i = \sum_j b_1^{x-x_{ij}} b_2^{k_j}$, where b_1 and b_2 could in principle be different constants. Several possibilities were tested, but $b_1 = b_2 = 2$ was found to be most efficient.
[45] G. Butland, J. M. Peregrim-Alvarez, J. Li, W. Yang, X. Yang, V. Canadien, A. Starostine, D. Richards, B. Beattie, N. Krogan, M. Davey, J. Parkinson, J. Greenblatt and A. Emili, Nature **453**, 531 (2005).
[46] S. Wassermann and K. Faust, *Social Network Analysis* (Cambridge University Press, Cambridge, England, 1994).
[47] The Zipf plot is related to the probability distribution of occurrence. If a subgraph i has rank R_i and occurs n_i times, then R_i is the number of graphs that occur n_i times or more. Hence $R_i = \int_{n_i}^{\infty} P(n)$, where $P(n)$ is the probability for a subgraph to occur n times. For a power law distribution $P(n) \propto n^{-\gamma}$ so $R_i \propto n_i^{1-\gamma}$, or $n_i \propto R_i^{\frac{1}{1-\gamma}}$.